

NSLS-II Fast Orbit Feedback with Individual Eigenmode Compensation



Yuke Tian

On Behalf of NSLS-II FOFB Team

Photon Science Directory

Brookhaven National Lab



2013 APS Workshop On Next-Generation Fast Orbit Feedback System For Storage Rings



Outline

- **NSLS-II orbit feedback system**
 - Technical requirements and specifications
 - NSLS-II site field vibration measurement
 - Slow and fast corrector locations
- **Fast orbit feedback with individual eigenmode compensation**
 - Typical fast orbit feedback algorithm
 - NSLS-II FOFB algorithm with individual eigenmode compensation
 - Comparison of the two algorithms
 - Solving the calculation problems from two directions
- **NSLS-II fast orbit feedback implementation**
- **Summary**

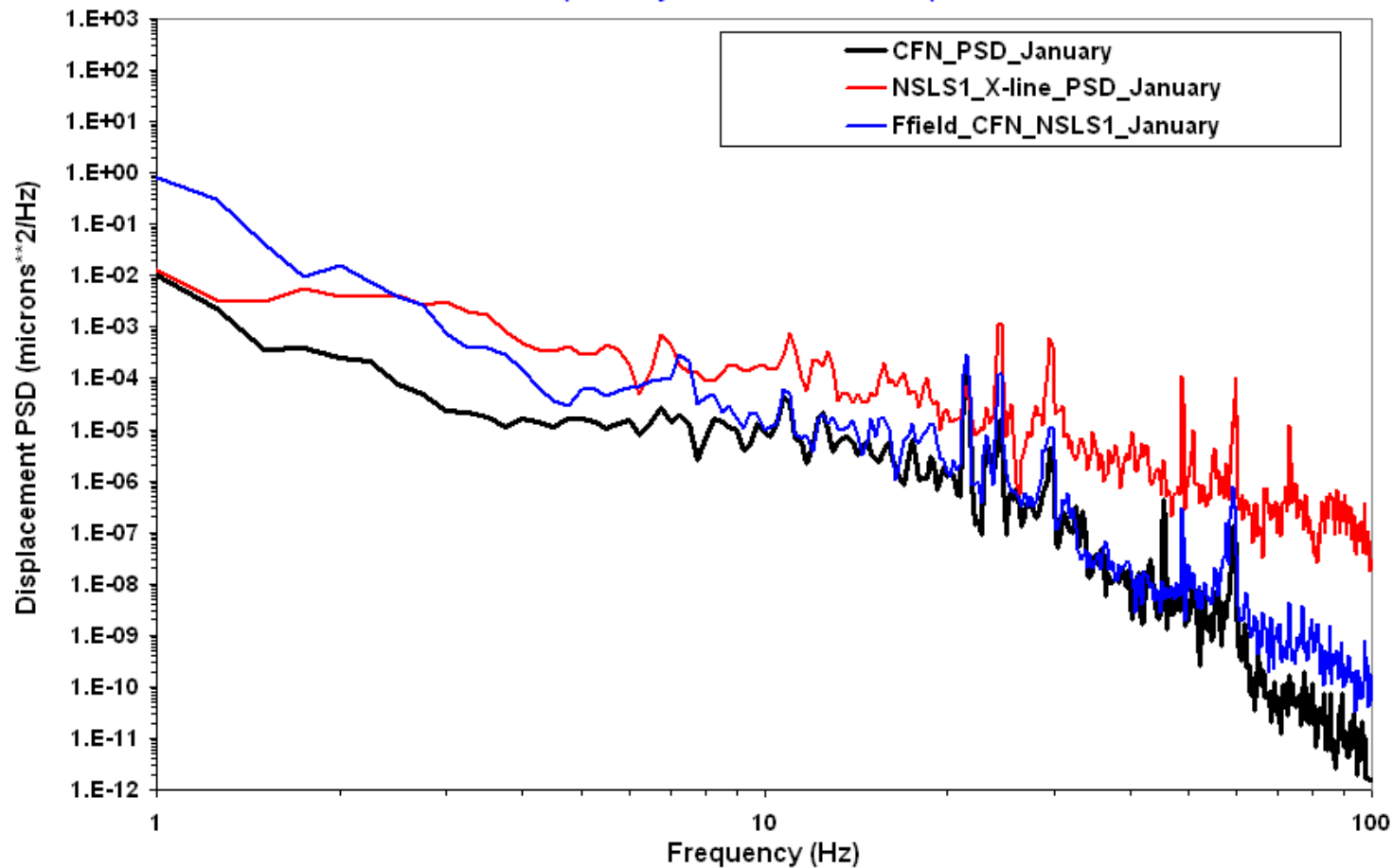
NSLS-II technical requirements & specifications

Energy	3.0 GeV	Energy Spread	0.094%
Circumference	792 m	RF Frequency	500 MHz
Number of Periods	30 DBA	Harmonic Number	1320
Length Long Straights	6.6 & 9.3m	RF Bucket Height	>2.5%
Emittance (h,v)	<1nm, 0.008nm	RMS Bunch Length	15ps-30ps
Momentum Compaction	.00037	Average Current	300ma (500ma)
Dipole Bend Radius	25m	Current per Bunch	0.5ma
Energy Loss per Turn	<2MeV	Charge per Bunch	1.2nC
		Touschek Lifetime	>3hrs

NSLS-II site field vibration measurement

Vertical PSD for NSLS1 X-Line; Free-Field between NSLS1 & CFN and
CFN Floor

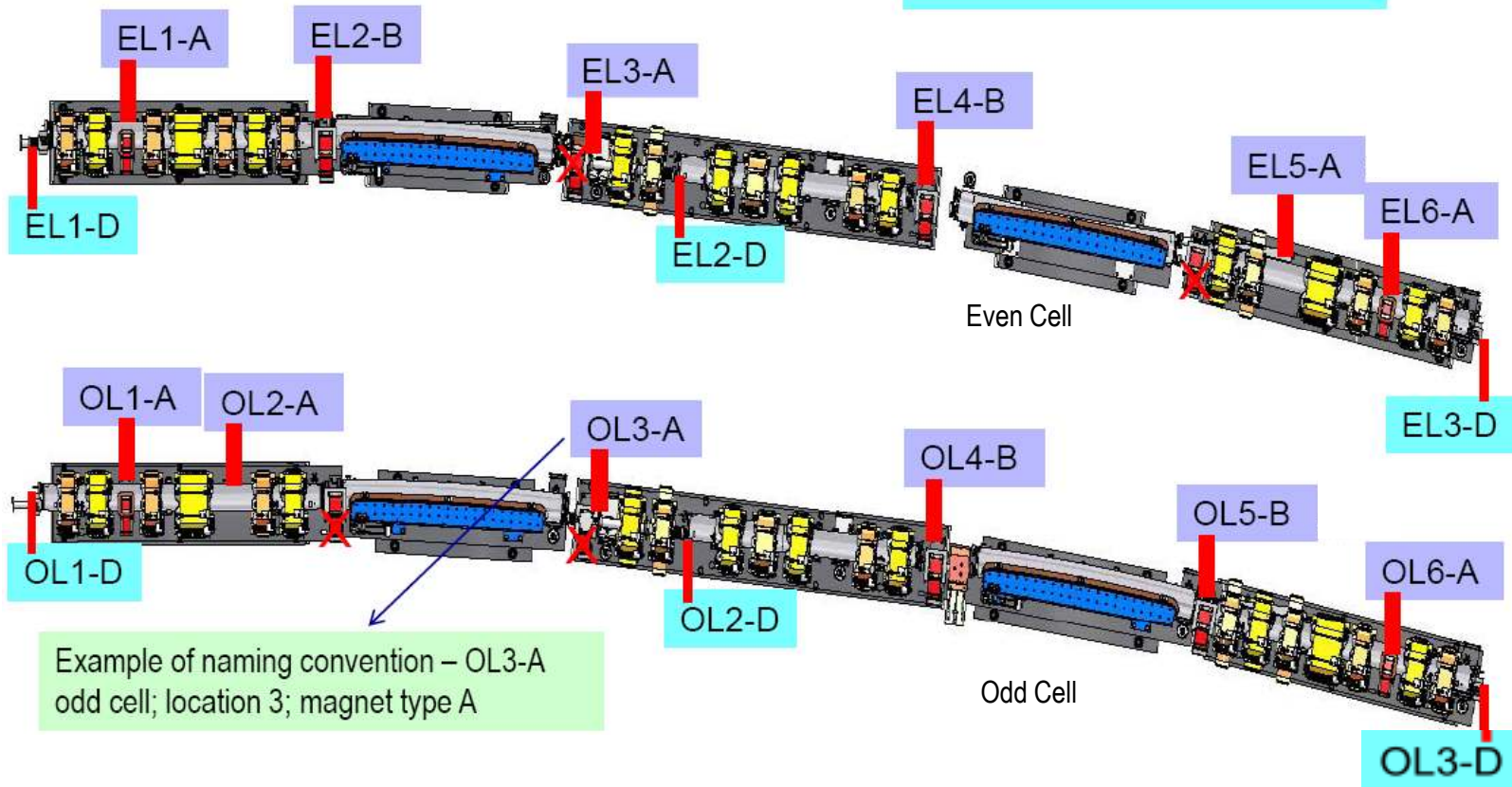
(January 2007 Measurements)



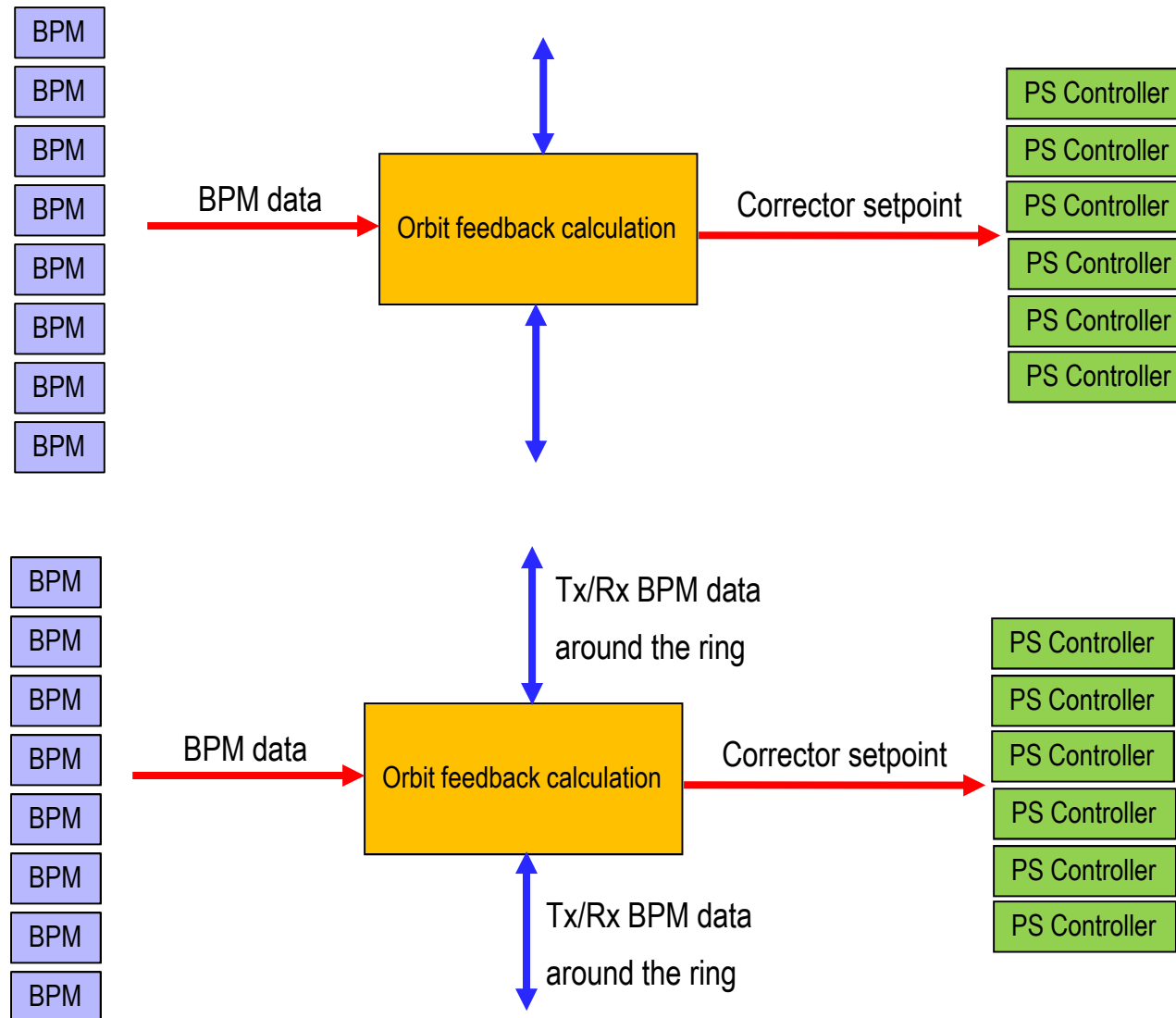
Slow and fast corrector locations

- A & B – Slow corrector; FS DC strength = 800 microrad
- A -100 mm Aperture (qty=8);
- B – 156 mm Aperture (qty=4); mounted over bellows

- D – Air core fast correctors; qty=6
- Mounted Over SS chamber
- FS DC Strength = 10-15 microrad
- Combined DC/AC function



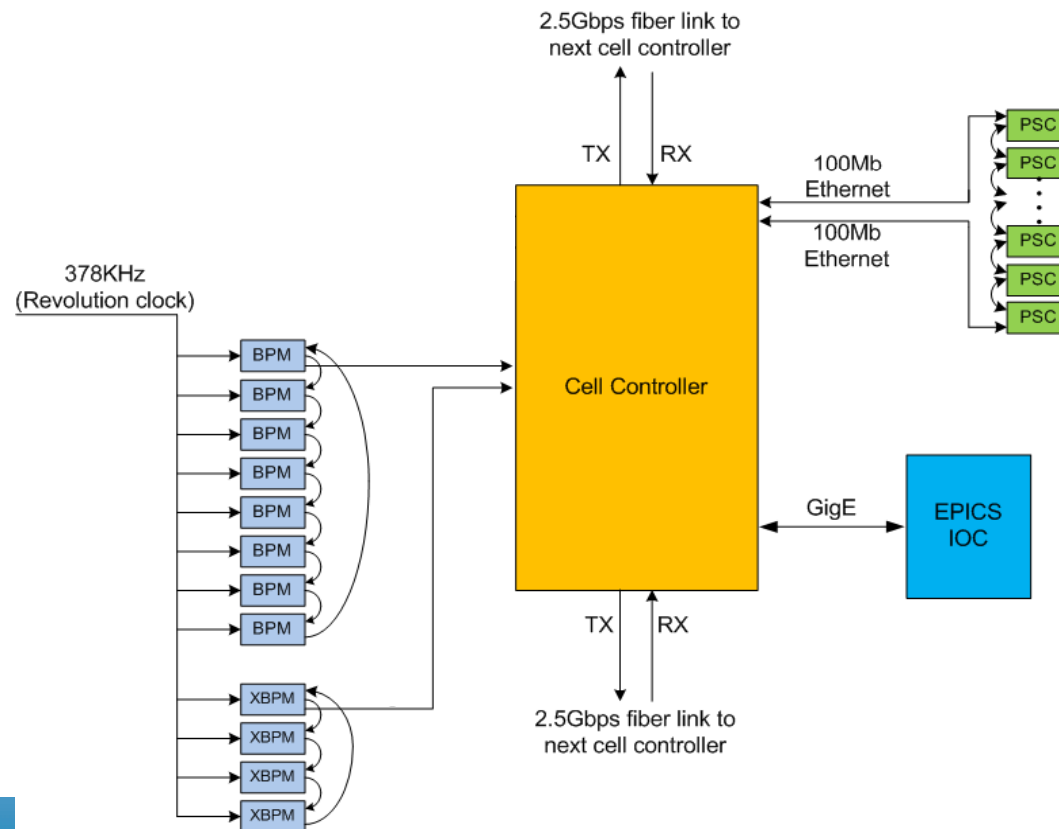
Typical Structure of a FOFB system



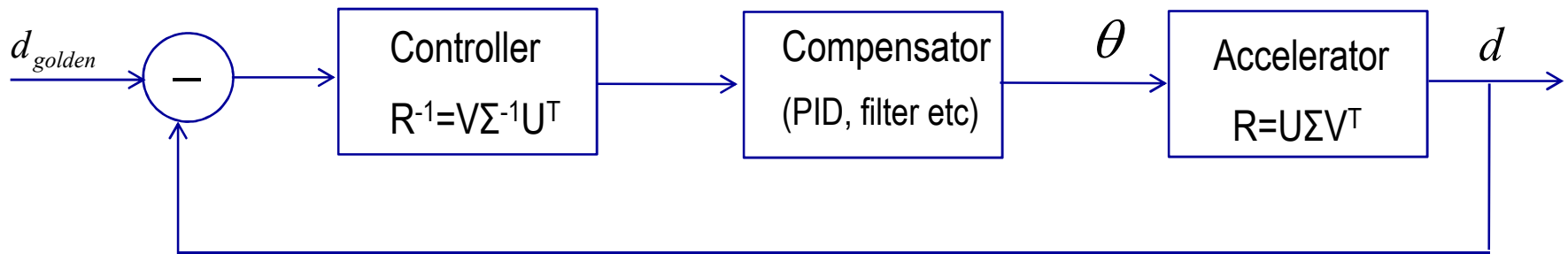
Can we make a clean architecture ?

1. Try to deliver BPM data to a place that orbit calculation module have directly access. Don't shove data around too much.
2. Similarly, try to deliver corrector setpoint from a place that orbit calculation module have directly access.
3. Looks like we need a place that can:
 - Receiver local BPM data;
 - Send/Receive BPM data to/for other cell;
 - Carry out FOFB calculation;
 - Send corrector setpoints to PS control system.

Cell Controller !



Typical fast orbit feedback algorithm



$$R_{M \times N} \bullet \theta_{N \times 1} = d_{M \times 1} \quad \text{R: response matrix}$$

$$\theta_{N \times 1} = R^{-1}_{N \times M} \bullet d_{M \times 1} \quad \text{R}^{-1}: \text{reverse response matrix}$$

The ill-conditioned response matrix will cause numerical instability.

Solution: 1) Truncated SVD (TSVD) regularization
2) Tikhonov regularization

Typical fast orbit feedback algorithm

$$R^{-1} = VD\Sigma^{-1}U^T = [\tau_1, \tau_2, \dots, \tau_N] \begin{bmatrix} \frac{1}{\sigma_1} \cdot \frac{\sigma_1^2}{\sigma_1^2 + \alpha} & & & \\ & \frac{1}{\sigma_2} \cdot \frac{\sigma_2^2}{\sigma_2^2 + \alpha} & & \\ & & \ddots & \\ & & & \frac{1}{\sigma_N} \cdot \frac{\sigma_N^2}{\sigma_N^2 + \alpha} \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \vdots \\ \delta_N \end{bmatrix}$$

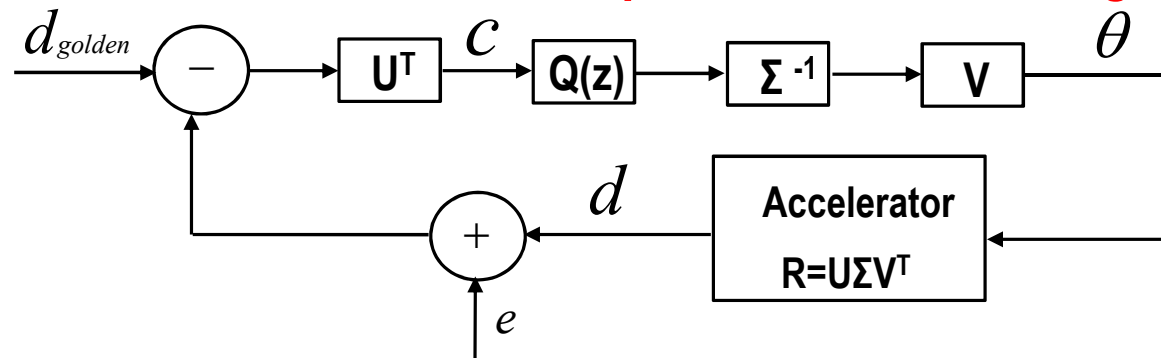
$$\theta_{Nx1} = R^{-1}_{NxM} \bullet d_{Mx1} \xrightarrow{\text{For each corrector plane}} \theta_i = R^{-1}_{i^{th} row} \bullet d_{Mx1}$$

Each of the corrector setpoint is calculated from a 1xM vector and Mx1 vector multiplication. If we don't exclude the X/Y coupling, M is the twice of the BPM number. The N is the twice of the corrector number. For NSLS-II, M=180*2=360, N=90*2=180. The above calculation is: 360 MAC (multiplication and accumulation).

NSLS-II FOFB algorithm – compensation for each eigenmode

- Fast orbit feedback system is a typical multiple-input and multiple-output (MIMO) system. For NSLS-II, there are 360 BPM readings and 180 fast corrector set points. The BPMs and correctors are coupled together. One BPM reading is the results of many correctors. One corrector kick can also affect many BPM readings. It is difficult to design a compensator for all noises with different frequencies.
- It is desirable if we can decouple the BPM and corrector relationship so that the MIMO problem can be converted into many single input single output (SISO) problems, for which control theory has many standard treatments.
- Fortunately, SVD already provides a solution: it projects the BPMs input into the eigenspace, where each component is independent. We can design many SISO type compensators (one for each eigenmode) and apply the standard SISO control theory to treat each eigenmode problem in frequency domain without affecting other eigenmodes.

NSLS-II FOFB calculation – compensation for each eigenmode



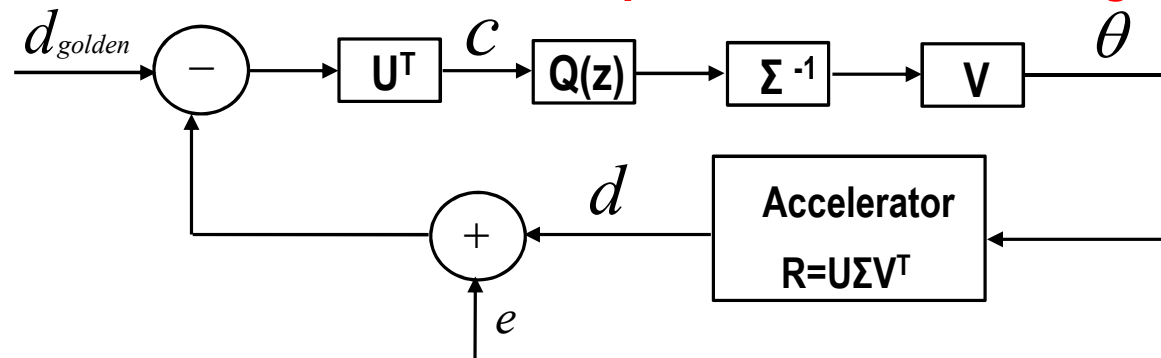
$$Q(z) = \begin{bmatrix} Q_1(z) & 0 & 0 & 0 \\ 0 & Q_2(z) & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & Q_N(z) \end{bmatrix}$$

c_1, c_2, \dots, c_N is the input projections in the eigenspace.

$Q_1(z), Q_2(z), \dots, Q_N(z)$ is the compensator for each eigenmode.

We want to prove that $Q_1(z), Q_2(z), \dots, Q_N(z)$ only corrects the corresponding eigenmode in eigenspace without affecting other eigenmodes.

NSLS-II FOFB calculation – compensation for each eigenmode



In cycle n ,

$$c(n) = U^T (d(n) + e(n)) \quad \theta(n) = V \Sigma^{-1} Q(z) c(n)$$

$$d(n+1) = U \Sigma V^T (V \Sigma^{-1} Q(z) c(n)) + e(n+1)$$

Since, $V V^T = V^T V = I \quad U^T U = I$

$$d(n+1) = U Q(z) c(n) + e(n+1) \quad c(n+1) = U^T d(n+1) = Q(z) c(n) + U^T e(n+1)$$

$$\begin{bmatrix} c_1(n+1) \\ c_2(n+1) \\ \dots \\ c_N(n+1) \end{bmatrix} = \begin{bmatrix} Q_1(z) & 0 & 0 & 0 \\ 0 & Q_2(z) & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & Q_N(z) \end{bmatrix} \begin{bmatrix} c_1(n) \\ c_2(n) \\ \dots \\ c_N(n) \end{bmatrix} + U^T e(n+1)$$

The pure effect of $Q_i(z)$ is on the i -th eigenvector.

The noise signals are also decoupled into the eigenspace.

This gives us freedom to suppress the noises in eigenspace using SISO control theory.

Eigenspace decompositions or not ? Calculation compare

Without eigenspace decomposition:

$$\theta_{Nx1} = Q(z)R^{-1}_{NxM} \bullet d_{Mx1}$$

For each corrector plane, M multiplication and accumulation(MAC), followed by k MAC for compensations.

Calculations: One corrector plane : M+k

For NSLS-II, M=360, N=180, k=3 (PID):

Calculation Amount (MAC)	Without eigenspace decomposition
One corrector strength	M+k =363

Decomposition or not ? Calculation compare

With eigenspace decompositions:

$$\mathbf{c}(n) = \mathbf{U}^T (\mathbf{d}(n) + \mathbf{e}(n))$$

$$\theta(n) = \mathbf{V} \Sigma^{-1} \mathbf{Q}(z) \mathbf{c}(n)$$

Calculations:

- 1) $N \times M$ MAC to project the inputs into eigenspace (decompositions).
- 2) $N \times k$ MAC for N compensators in each eigenmode.
- 3) N MAC to get one corrector strength.

For NSLS-II, $M=360$, $N=180$, $k=3$:

Calculation Amount (MAC)	With eigenspace decomposition
One corrector strength	$N(M+k+1) = 65,520$

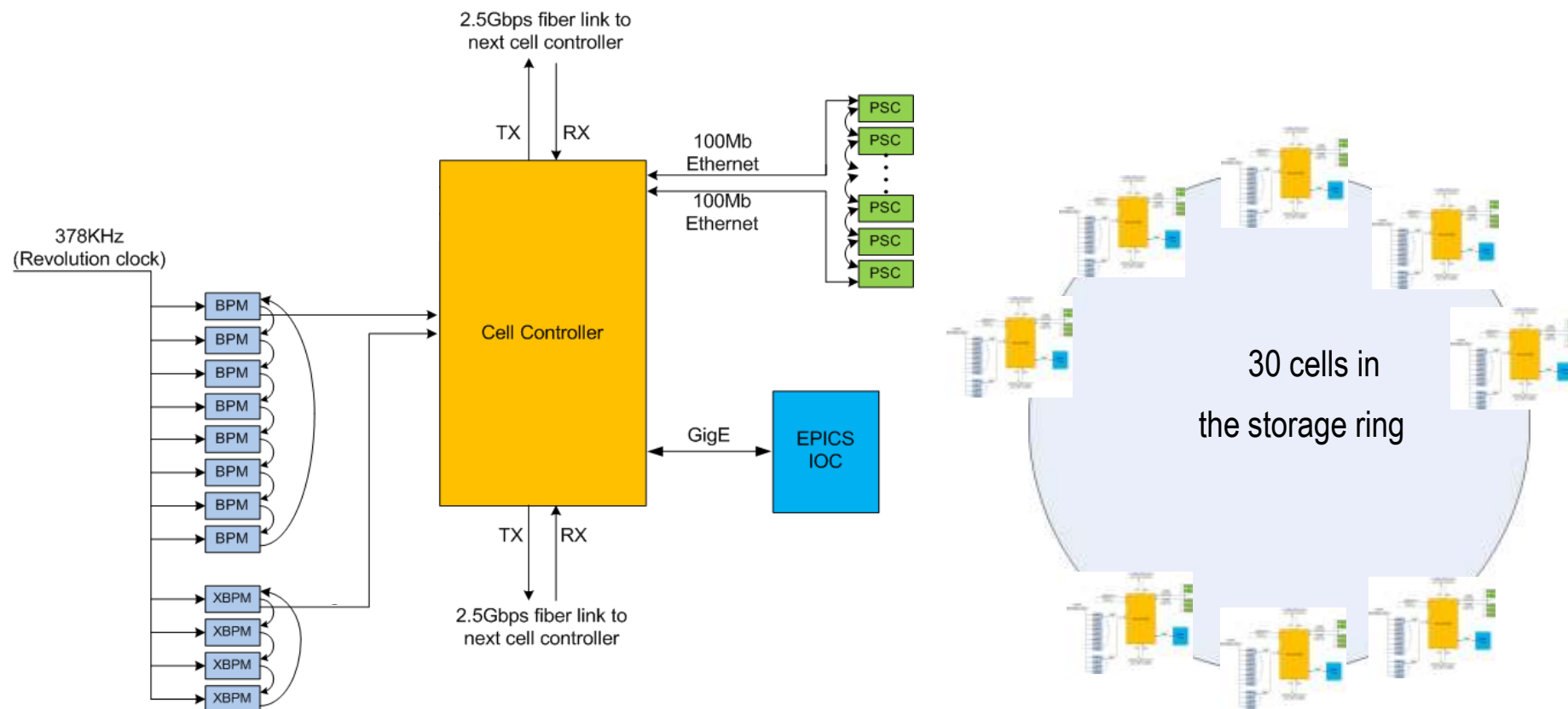
Assume 50us of 10Khz FOFB time is used for the calculations:

One corrector strength calculation: $65,520/50\mu\text{s} = 1.3 \text{ GMAC/s}$ (1.3billion MAC per second).

The high end DSP chip gives about 200-300 Millions floating point operations (MFLOP).

Solving the calculation problems from two directions

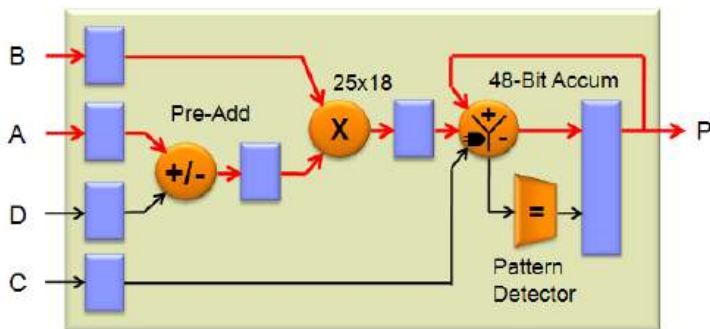
Distributed FOFB system: two tier communication



All the BPM data is delivered to all the 30 cell controller within 12us, cell controller only needs to calculate the local corrector strength. This distributed architecture reduces the calculation by factor of 30.

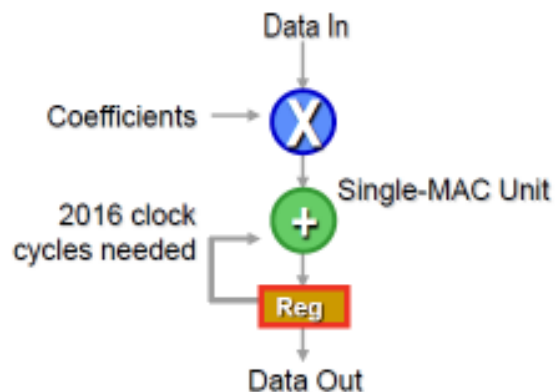
Solving the calculation problems from two directions

FPGA's powerful DSP performance

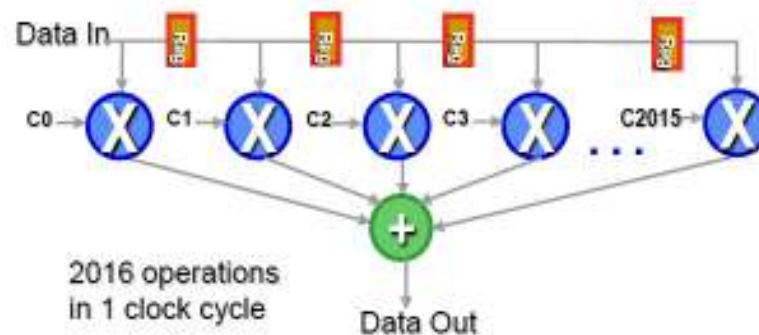


One DSP48E1 block in FPGA

FPGA's DSP power is mainly gained through the parallel computation capability. The parallelism increases FPGA DSP power (vs generic DSP or CPU) by a factor of more than 2000 (Virtex 6).



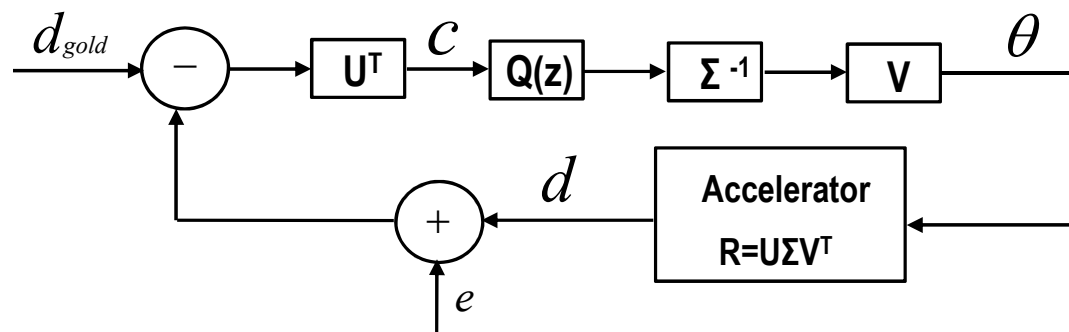
Standard DSP/CPU process: sequential



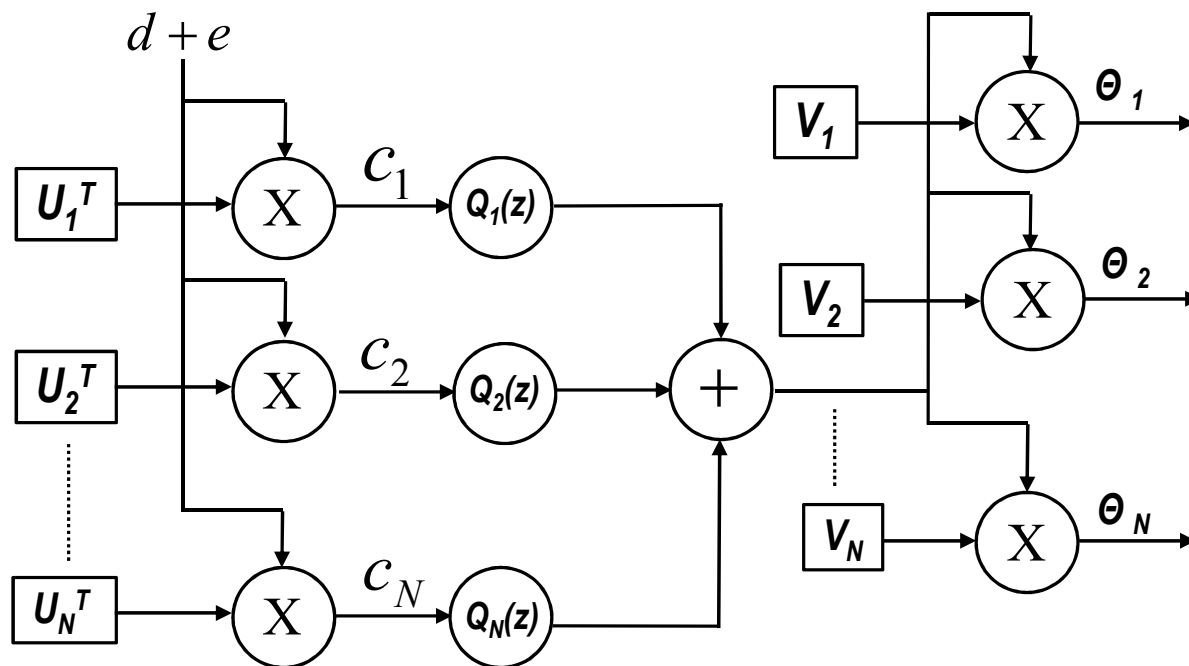
FPGA's fully implementation

Solving the calculation problems from two directions

FOFB calculate in FPGA:

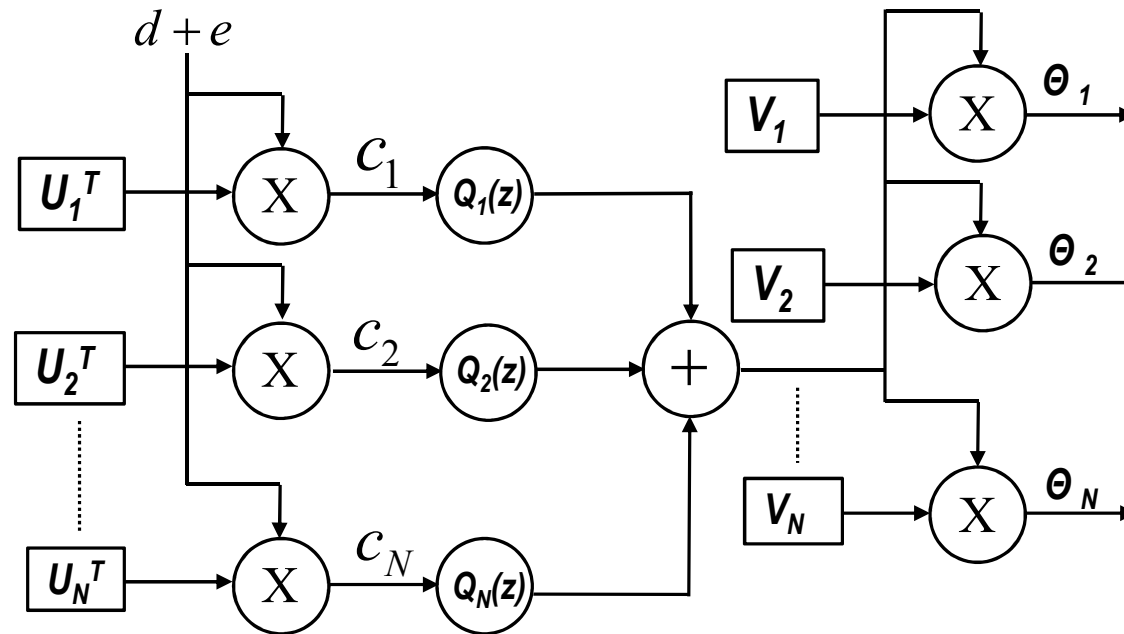


NSLS-II FOFB Model



Solving the calculation problems from two directions

FOFB calculate amount in FPGA:



Eigenspace decomposition: $c(n) = U^T (d(n) + e(n))$

All components $c_1(n), c_2(n), \dots, c_N(n)$ are calculated in parallel (M MAC)

Compensation for each eigenmode: $Q(z)c(n)$

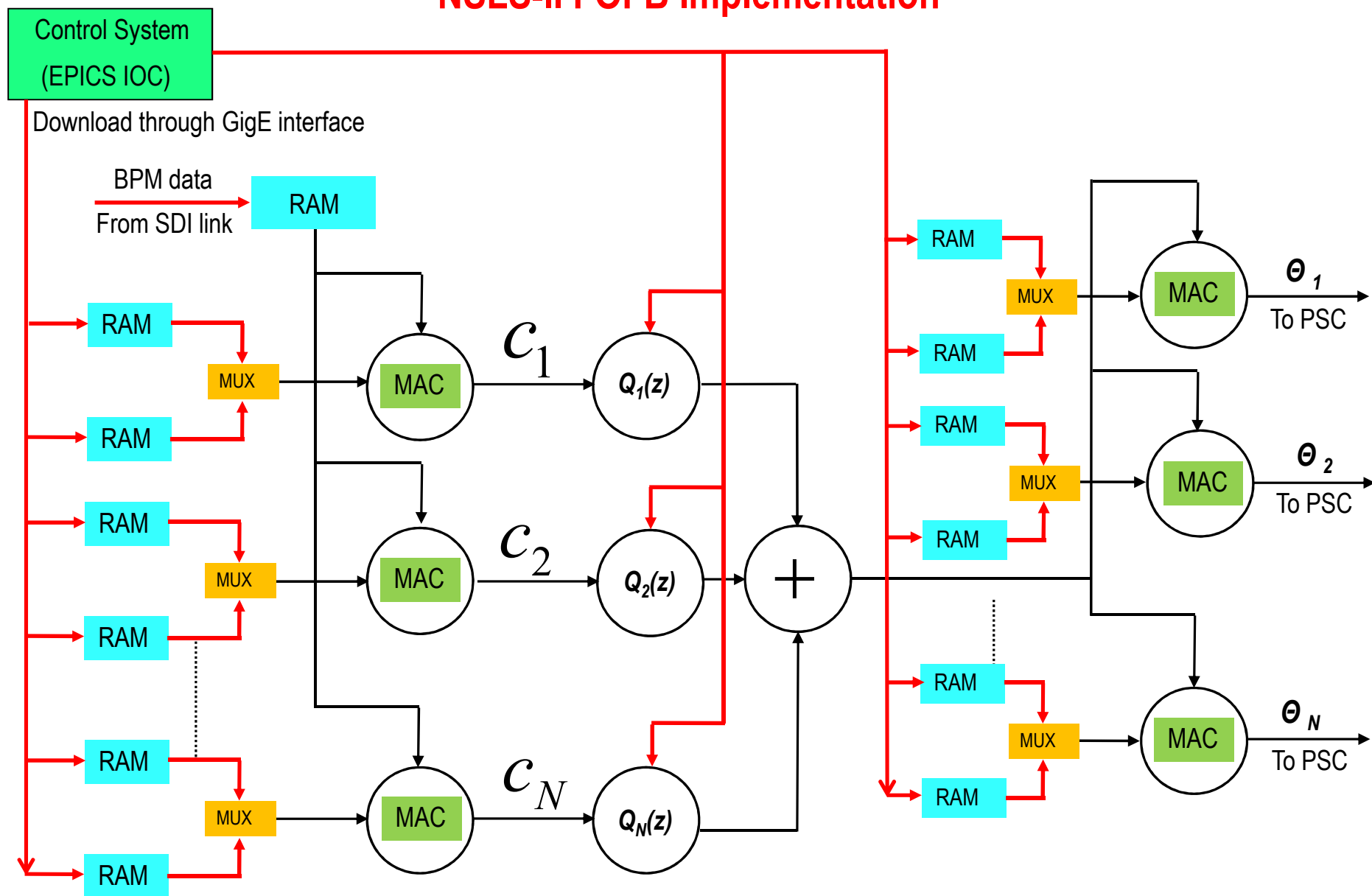
The k step compensations are done in parallel. (k MAC)

Corrector strength calculation: $\theta(n) = V\Sigma^{-1}Q(z)c(n)$

All corrector strength is calculated in parallel: N MAC

Each parallel calculation: $M+N+k = 360 + 3 + 180 = 543$ MAC. FPGA will finish it within a few us.

NSLS-II FOFB Implementation



NSLS-II FOFB Special Case Considerations

$$R_{M \times N} \cdot \theta_{N \times 1} = d_{M \times 1}$$

R: response matrix. M:BPM readings; N:corrector plane numbers

Fast orbit feedback system needs to be able to handle some special cases.

→ If one (or a few) BPM is too noisy and we don't want to use its data in the fast orbit feedback calculation, we can remove the corresponding row in the response matrix $R_{M \times N}$. The high level application program can re-calculate the SVD matrixes and download them to the cell controller. The cell controller will use the new SVD matrixes in its calculation.

→ Similarly, if one (or a few) fast corrector is down and we don't want to use it in the fast orbit feedback, we can remove the corresponding column in the response matrix $R_{M \times N}$. The high level application program can re-calculate the SVD matrixes and download it to the cell controller. The cell controller will use the new SVD matrixes in its calculation.

→ If we need to change the whole feedback algorithm, we can always re-configure the FPGA in the cell controller through Ethernet. This takes a few minutes.

NSLS-II Fast Orbit Feedback Status

- The hardware design (PCB/assembly/testing, chassis) for BPM, cell controller and PSC are all done. The production units are being installed in the storage ring.
- BPM and PSC FPGA firmware, EPICS drivers and database development are all done.
- All cell controller blocks (SDI, FOFB etc) are all done. The cell controller integration is in progress.
- Since we have the fast fiber SDI to deliver data around the ring, cell controller's SDI link will also be used as fast machine protection system that deliver critical system (such as the vast valve signal from vacuum system) around the ring within much less than 1ms. This latency is impossible for PLC to achieve.

Hardware

- We like to open the hardware we designed at BNL to the community. These includes all the PCB design, the FPGA firmware design and the EPICS IOC (driver, database) design. We welcome any suggestions and cooperation from the community.

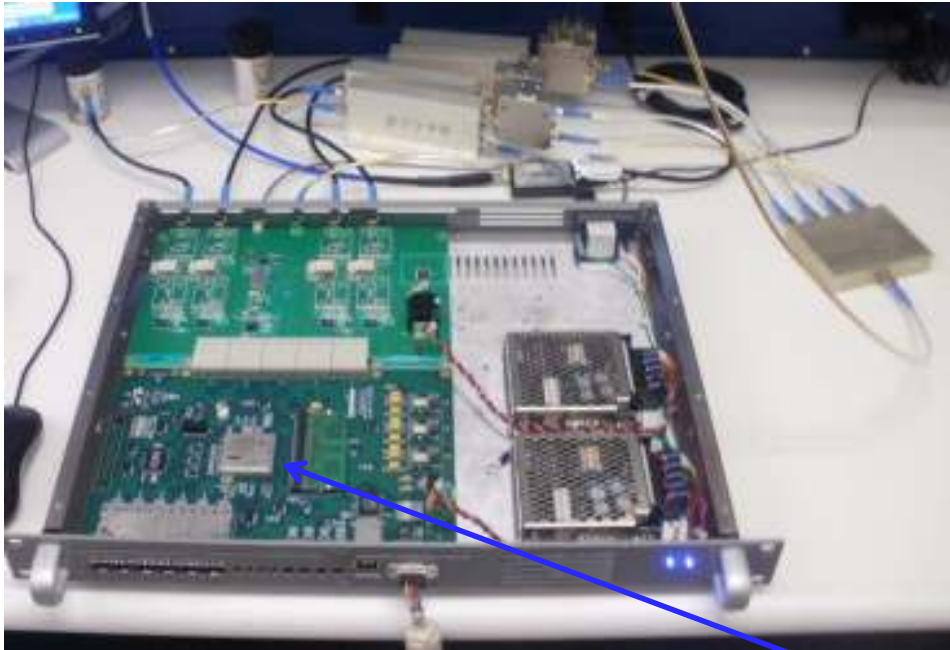
PSC



PSI

Hardware

- The digital front end (DFE) board (Virtex6 FPGA) can be used as a common platform for many systems since it provides the basic common elements such as large memory (2 GB DDR3), embedded CPU, Gigabit Ethernet port to EPICS IOC, high speed SDI link (use 2 of the 6 available SFPs), and many user IOs (>160) to the user daughter board.



BPM



Cell Controller

Share the same DFE

Hardware

100Migabit/s link for corrector setpoints

IO signals (16 inputs, 12 ouptuts, 4 Vout) for fast machine protection

IO board

2 GB DDR3

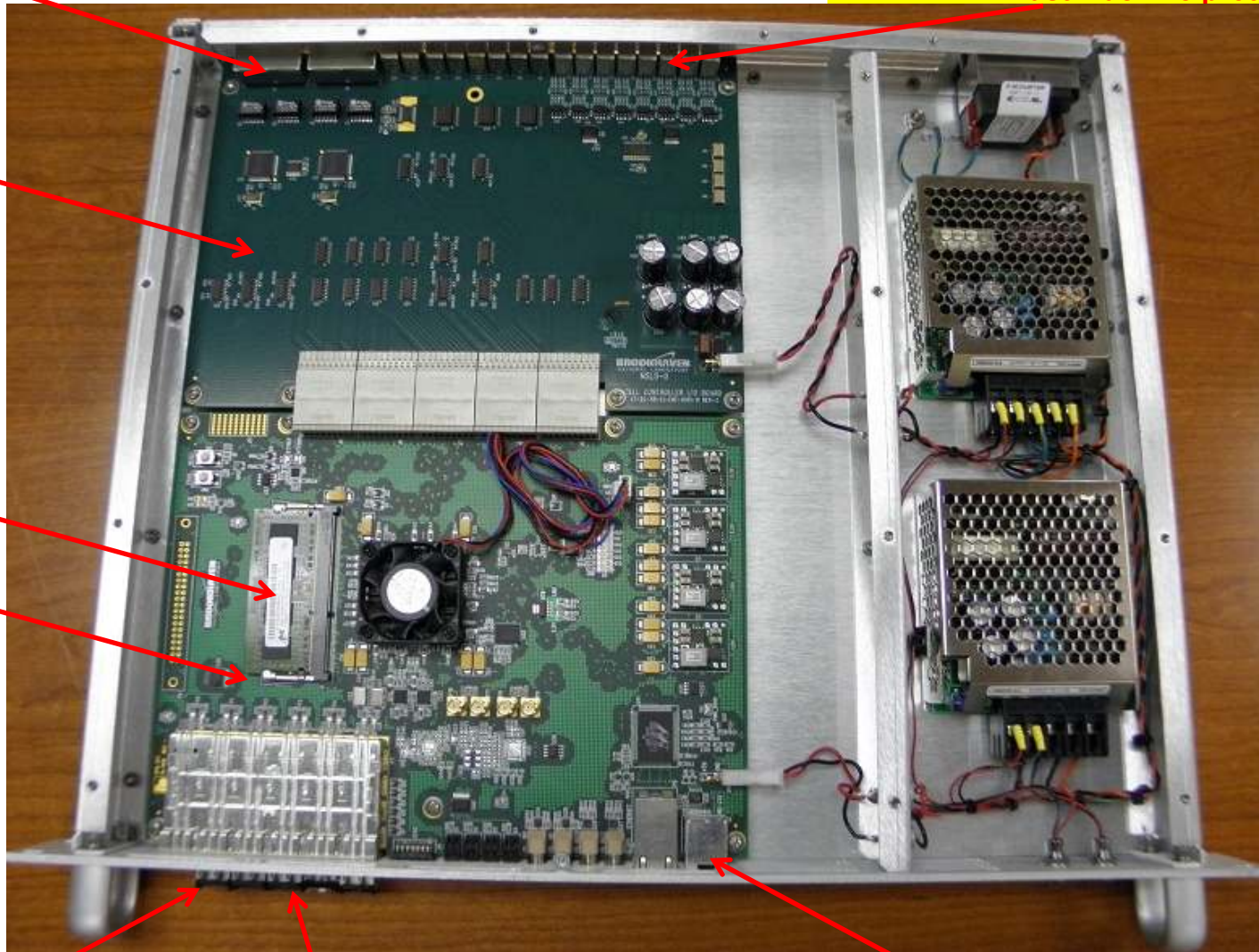
DFE

Embedded Event Received

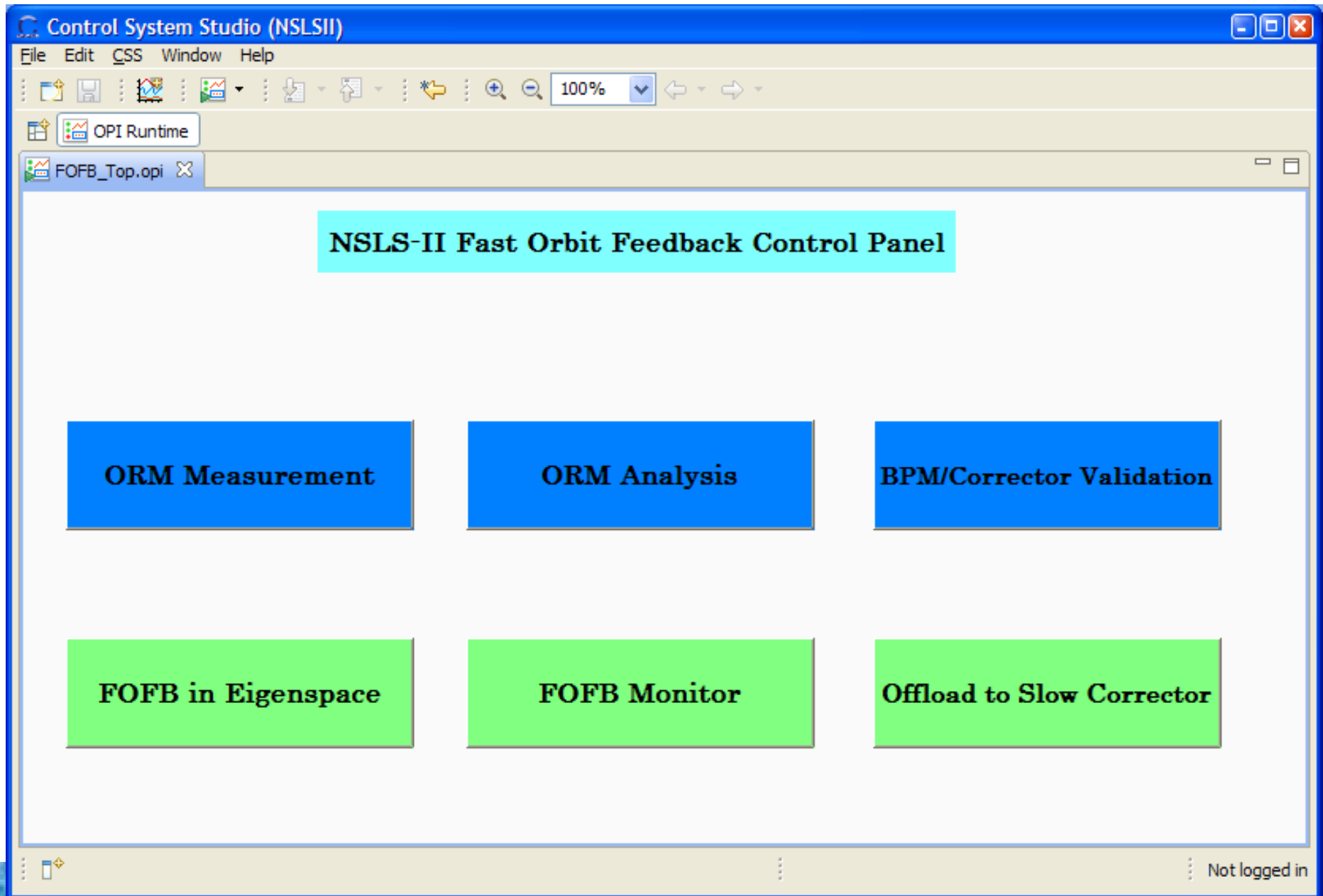
2-5 Gigabit/s SDI link for BPM data

Cell Controller

Gigabit Ethernet to EPICS IOC



NSLS-II FOFB Tasks Control Panel



Summary

- NSLS-II's stringent emittance requirements need an efficient fast orbit feedback system. The two tier communication structure and the FPGA-based fast orbit feedback calculation architecture is designed for achieve the requirements.
- Algorithm with individual eigenmode compensation is proposed. The typical MIMO feedback problem is converted into many SISO problems. This algorithm enables us to correct the beam orbit in eigenspace.
- We compared the calculations for FOFB with and without individual eigenmode compensation. We found that the proposed NSLS-II FOFB algorithm needs a large amount of calculations. However, benefited from NSLS-II FOFB architecture, the challenge can be conquered.
- We expect a successful application of the NSLS-II FOFB algorithm during the NSLS-II commissioning and daily operation.

Acknowledgments

- FOFB algorithm
Lihua Yu
Sam Krinsky
- BPM/ Cell controller development :
Om Singh
Kurt Vetter
Joseph Mead
Kiman Ha
Alfred Dellapenna
Joseph De Long
Bob Dalesio
- PSC and PS design:
Wing Louie
John Ricciardelli
George Ganetis